

OCR
A Level
Computer
Science
H446 – Paper 1

3

**Thinking
procedurally**

Unit 10
Computational
thinking



PG ONLINE

Objectives

- Identify the components of a problem
- Identify the components of a solution to a problem
- Determine the order of the steps needed to solve a problem
- Identify sub-procedures necessary to solve a problem

Thinking procedurally

- Most problems of any size need to be broken down into their component parts
- Think of the problem:
“How will I be able to go to University?”
- What are the different aspects of this problem?
- Is this a computational problem?
- Suggest other problems that would need to be decomposed into separate, smaller sub-problems

Decomposition

- Procedural decomposition means breaking a problem into a number of sub-problems, so that each sub-problem accomplishes an identifiable task
- The sub-problems may themselves be further subdivided



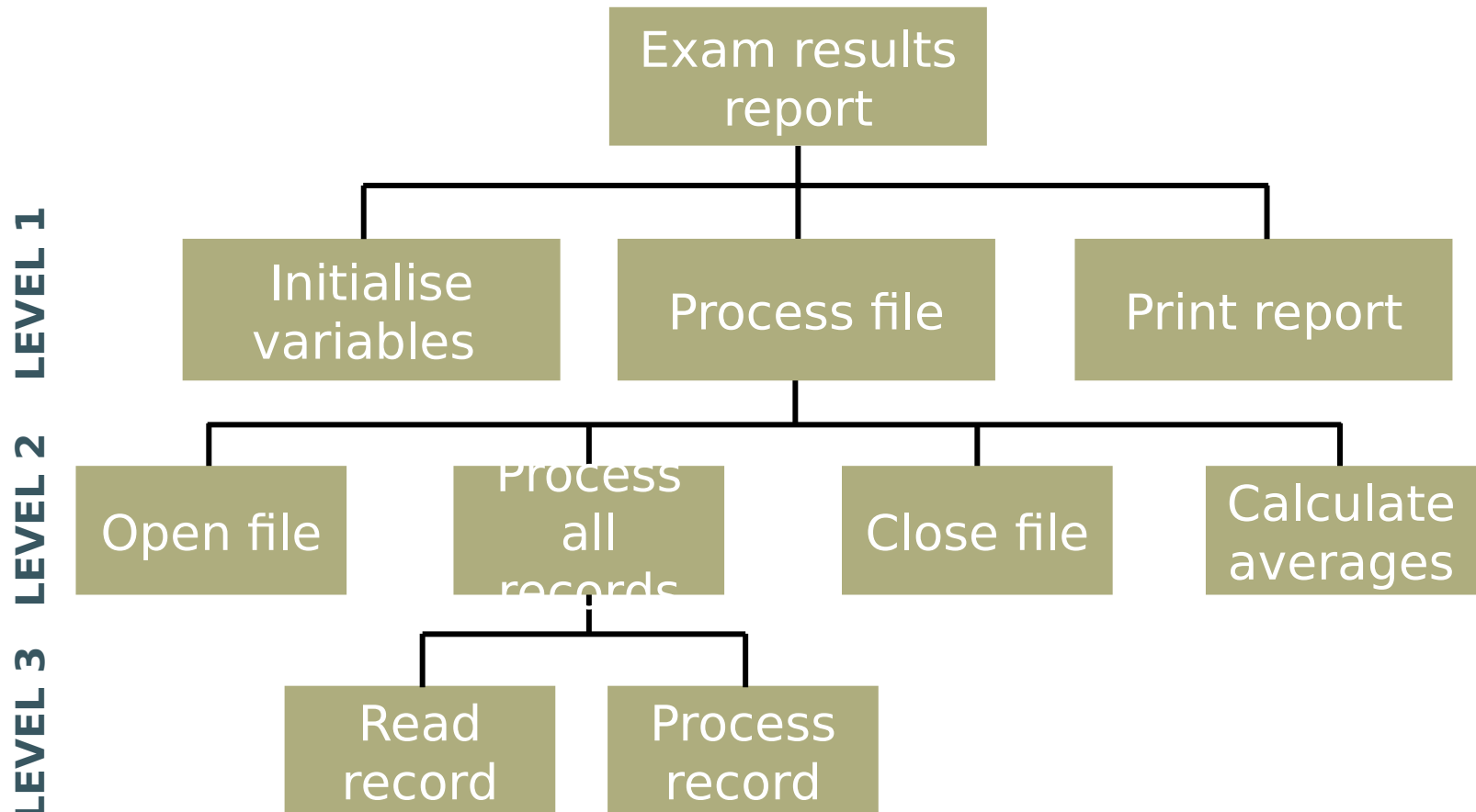
A definition of structured programming

- Structured programming aims to improve the clarity and quality of a program
- It is a method of writing a computer program which uses
 - **Modularization** for program structure and organisation, i.e. breaking the problem down into subroutines
 - **Structured code** for the individual modules – that is, code which uses the basic constructs of **sequence**, **selection** and **iteration**
 - **Recursion**

A top-down design model

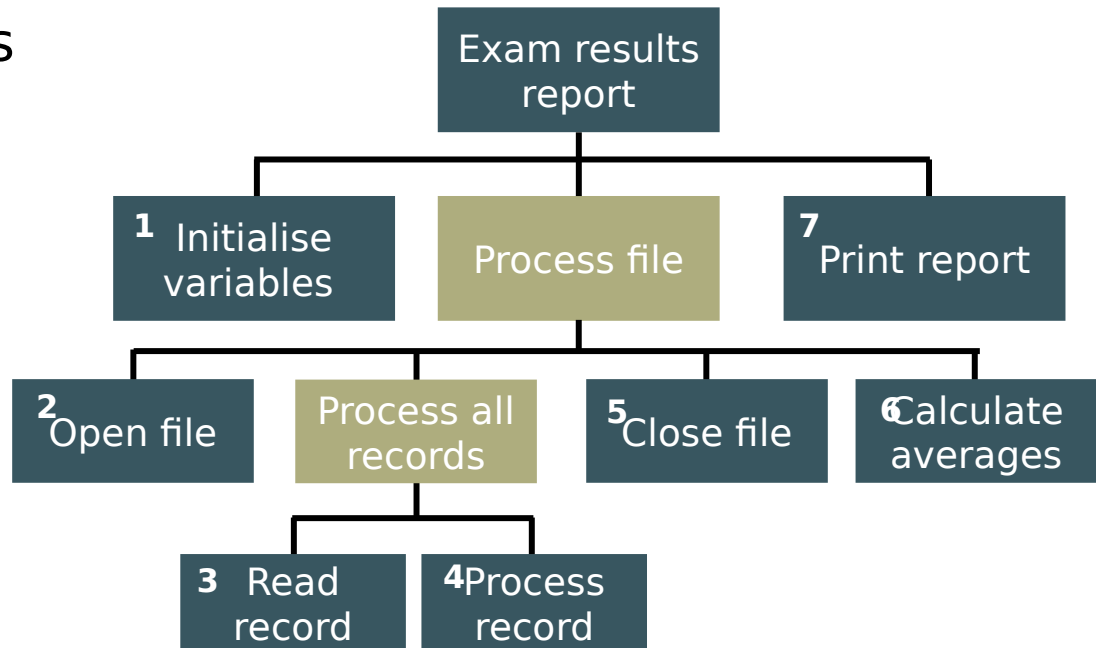
- Structured programming uses a top-down design technique
- A program is divided into sub-procedures, or modules, which are called from the main program
- Any of the sub-procedures may be further broken down into smaller sub-tasks, with the smallest performing a single function
- A **hierarchy chart** is often used to show the overall program structure

Hierarchy chart



Hierarchy chart

- Each logical process is broken down into smaller components until it cannot be broken down any further
- Execution takes place from left to right, always at the lowest level component
- Selection and iteration are not shown in a hierarchy chart
- Why are two of the boxes shown in a paler colour?



Benefits of modularisation

- Programs are more easily and quickly written
 - Large programs are broken down into subtasks/subroutines that are easier to program and manage
 - Each subroutine (i.e. module) can be individually tested
 - Modules can be re-used several times in a program
 - Frequently used modules can be saved in a library and used by other programs
 - Several programmers can simultaneously work on different modules, shortening development time
- Can you think of some more benefits?



More benefits of modularisation

- Programs are more reliable and have fewer errors
 - It is easier to find errors in small self-contained modules
- Programs take less time to test and debug
- Programs are easier to maintain
 - A well-organised, modular program is easier to follow
 - It is easier to find which module needs to be changed
 - Self-contained modules mean that the change should not affect the rest of the program
 - New features can be added by adding new modules

Good programming practice

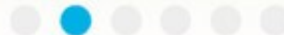
- Use meaningful identifiers (e.g. variable and procedure names)
- Define and document the inputs, outputs and preconditions for each sub-procedure
- Add lots of meaningful comments within the program
- At the bottom level, each sub-procedure should perform a single task
- Keep each sub-procedure self-contained by passing parameters and using local variables

Identifying the components

- Once you have identified the component parts of a problem, you can plan the overall method of solution
- This will involve writing **procedures** or **functions** and passing **parameters**
- For example, a car dealer might want a program to help potential customer select options for a new car
- This might use a procedure to display a particular model of a car, passing parameters for colour, number of doors, wheels etc.



Choosing car options



Back: Wheels

Engine

Wheels

Paint

Interior

Extras

Overview

Next: Interior

Black (Non
Metallic)
Standard



Tornado
Red (Non
Metallic)
Standard



Urano Grey
(Non
Metallic)
Standard



Pure White
(Non
Metallic)
£260



Carmen Red
(Metallic)
£540



Limestone
Grey
(Metallic)
£540



Night Blue
(Metallic)
£540

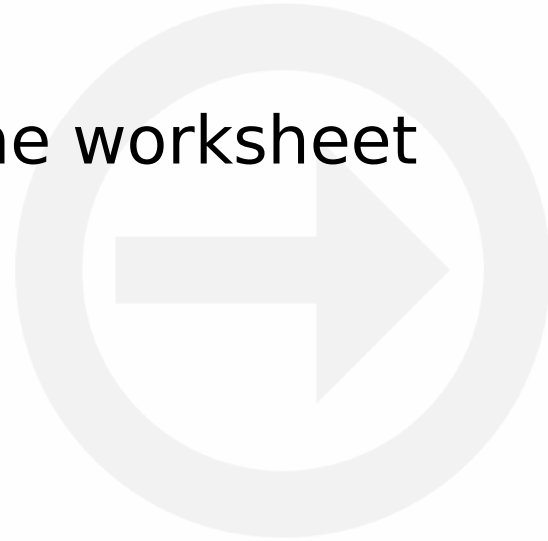


Pacific Blue
(Metallic)
£540



Worksheet 3

- Now try the questions in the worksheet



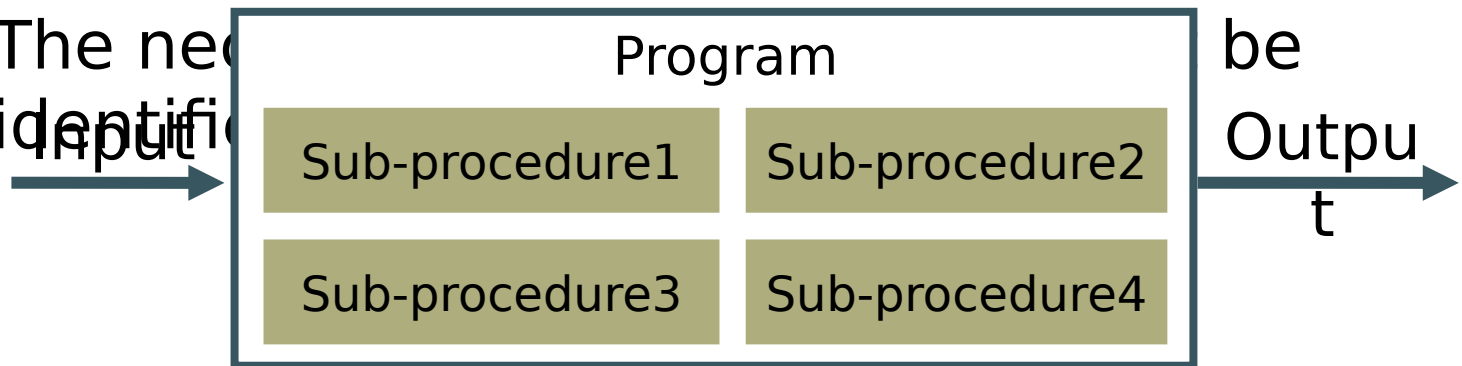
Modular programming

- Modular design and programming techniques are most useful for large, complex programs
- Some programs have thousands or even millions of lines of code
- In small programs of less than a page of code, it may not be worth writing individual modules for every subtask

out << "Hello, world!\n";

Plenary: Thinking procedurally

- The components of the **problem** need to be identified
- The components of the **solution** need to be identified
- The **order of the steps** needs to be determined
- The need to be identified



Copyright

© 2016 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.

